Module 3

Ananya Verma, Vikas Jadhav Y

February 2025

Activity 1

A Bernoulli random variable is a variable X that only takes two values such that,

$$p(X) = \begin{cases} p & X = 1 \\ 1 - p & X = 0 \end{cases} \tag{1}$$

We can model a coin toss as a Bernoulli variable by making the correspondence for heads and tails with $X = 1$ and $X = 0$ respectively.

In the case of N coin tosses, the data can be represented by the number of heads obtained. So if we obtain k heads in out N tosses, the likelihood of that happening is given by,

$$\mathcal{L}(k|p) = \binom{N}{k} p^k (1-p)^{N-k} \tag{2}$$

and hence the log likelihood is given by

$$log(\mathcal{L}(k|p)) = log\binom{N}{k} + klog(p) + (N-k)log(1-p) \tag{3}$$

The maximum likelihood estimate for p, p_{mle} is the value that maximises the likelihood or the log likelihood.

$$\frac{d\mathcal{L}(p_{mle})}{dp} = 0 \tag{4}$$

$$\frac{k}{p_{mle}} - \frac{(N-k)}{(1-p_{mle})} = 0 \tag{5}$$

$$p_{mle} = \frac{k}{N} \tag{6}$$

In our case, we set $p = 0.5$. For our trial we obtained $k = 47$, giving a measured value of the $\boxed{p_{mle} = \frac{47}{100} = 0.47}$

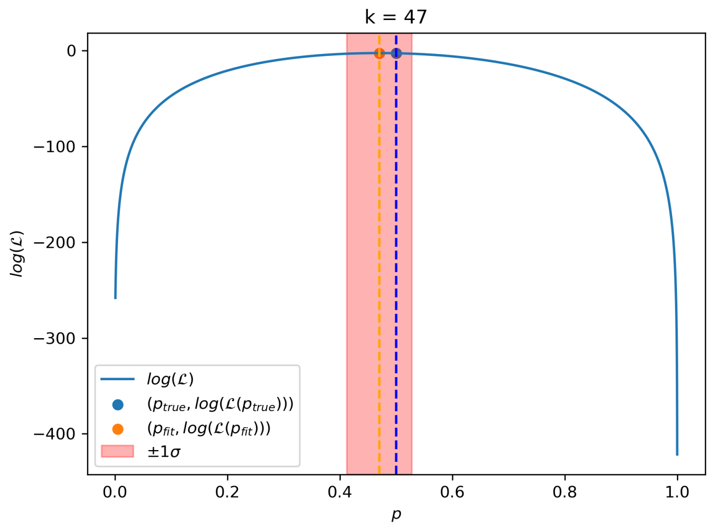

Figure 1: The plot of the log likelihood with p for $k = 47$

The uncertainities are given by p_σ such that

$$log(\mathcal{L}(p_{sigma})) = log(\mathcal{L}_{max}) - 0.5 \tag{7}$$

This comes out to be $\boxed{p_{sigma} = (0.4122589, 0.52828297)}$

Activity 2

We model the rate of the decay as follows

$$y(t) = w_1 e^{-s_1 t} + w_2 e^{-s_2 t} + w_3 t \tag{8}$$

However, since the time values go up to $t \sim 800s$, the computer will lose accuracy in calculating such small numbers. Hence, it is better to rescale the time as well as convert the exponential fit to a polynomial fit by making a change of variables

$$x \equiv e^{-t/100} \tag{9}$$

We can thus rewrite our decay rate in terms of x as,

$$y(x) = w_1 x^{\tilde{s_1}} + w_2 x^{\tilde{s_2}} - 100 w_3 log(x) \tag{10}$$

where $\tilde{s_1} = 100 s_1$ and $\tilde{s_2} = 100 s_2$.

We obtain the fit parameters and rescale them back to the parameters as defined in equation (8)

1. $w_1 = 979.82$

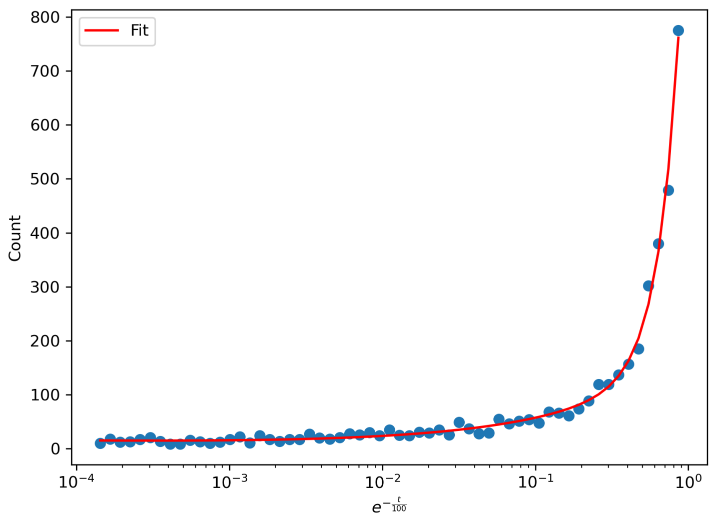

Figure 2: Fit of y vs $x \equiv e^{-t/100}$

2. $w_2 = 169.62$
3. $w_3 = 0.015$
4. $s_1 = 0.032$
5. $s_2 = 0.005$

The fit results are shown in Figure (2) and Figure (3)

We then calculated the pull of the fit at each time point. The pull is defined as

$$\text{pull} = \frac{(d_i - y(t_i))}{\Delta_i} \tag{11}$$

where d_i is the count at time t_i and Δ_i is the uncertainity in the count.

We then calculate the chi-square value of the fit

$$\chi^2 = \sum_i \frac{(d_i - y(t_i))^2}{y(t_i)} \tag{12}$$

For our fit, we obtain, $\boxed{\chi^2 = 79754.42}$

The degree of freedom in this case is the number of data points minus the number of parameters which is $\boxed{dof = 54}$

This gives us a $\boxed{p - value = 0.0}$ which is indicative of good fit.

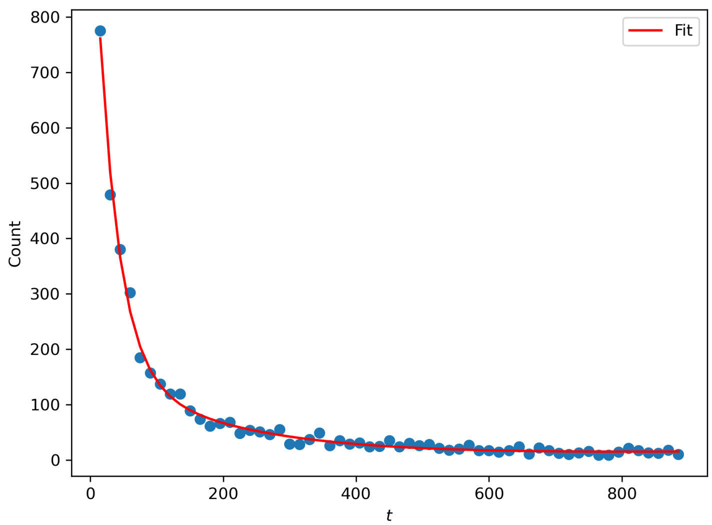

Figure 3: Fit of y and t

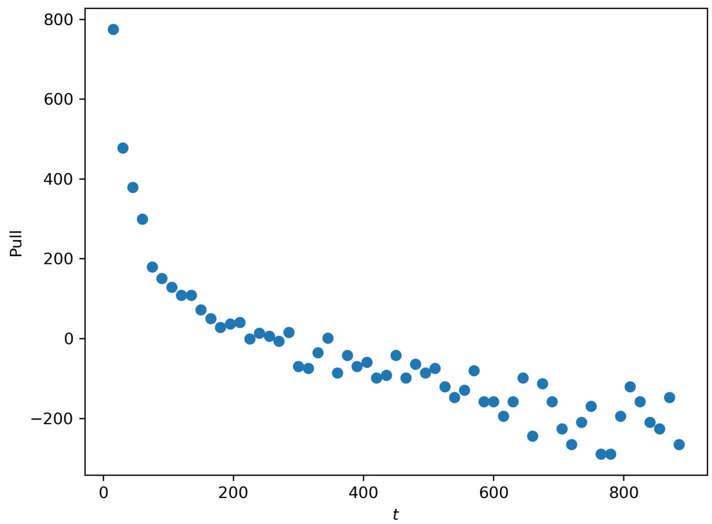

Figure 4: The Pull of the fit

Activity 3

Fitting Strategy

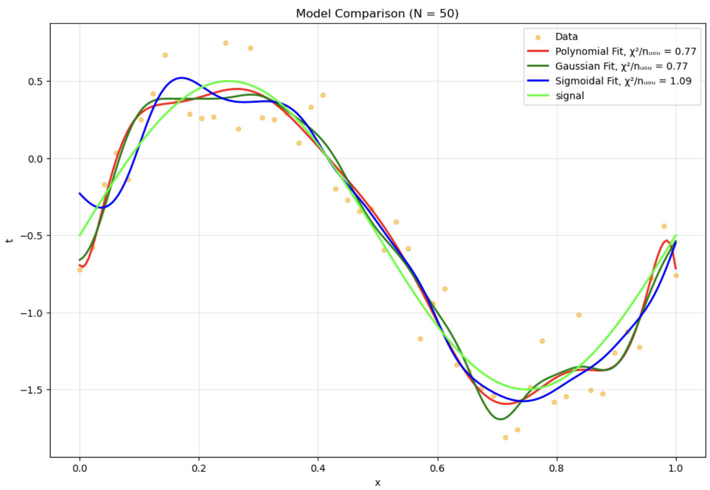

Figure 5:

The core approach employs a linear least squares method to fit the basis functions to our data. Rather than utilizing iterative optimization techniques, we implemented a direct solution method using `np.linalg.lstsq`. This method determines the optimal coefficients for our basis functions by minimizing the sum of squared residuals between model predictions and actual data points.
The process proceeds through several steps:

1. First, we construct a design matrix $\mathbf{\Phi}$ using our chosen basis functions (polynomial, Gaussian, or sigmoidal)

2. Then, we directly solve the normal equations using singular value decomposition (SVD)

3. This yields the optimal weights $\mathbf{w}$ that minimize the squared error between our model and the data

This approach is particularly efficient because for linear-in-parameters models (which includes our basis function models), we can find the exact solution without requiring iterative optimization. It is analogous to solving a system of equations directly rather than employing a trial-and-error approach until convergence.
The method also handles potential numerical issues effectively — if our basis functions are nearly linearly dependent (which can occur with high-degree polynomials), the SVD approach will still provide a stable solution.

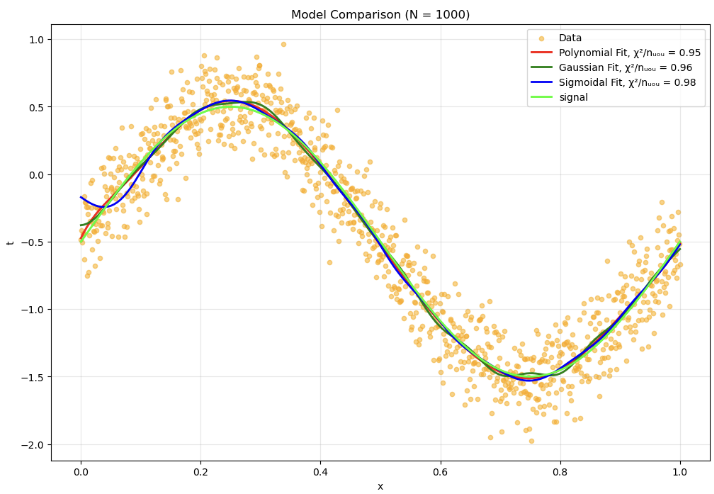

Figure 6:

For the Bayesian extension, we augment this framework by incorporating prior knowledge about the weights and uncertainty in our measurements. This yields not merely a single best fit, but a distribution over possible fits, which proves particularly valuable in scenarios with limited data or noisy measurements.

Analysis of Results and Strategies for Reliable Predictive Modeling

Analyzing our results across different basis functions and sample sizes reveals several key insights:

Comparative Analysis of Basis Functions

When comparing polynomial, Gaussian, and sigmoidal basis functions, each exhibits distinct characteristics affecting their predictive reliability. Polynomial basis functions demonstrate strong global behavior capture but suffer from edge effects, particularly with higher degrees. Gaussian basis functions excel at capturing local features and show enhanced stability at boundaries. Sigmoidal basis functions prove particularly effective at modeling transitions and maintain superior numerical stability.

Impact of Sample Size

Sample size emerges as a crucial determinant of model reliability. With small samples ($N = 10$), we observe:

- Large uncertainty bands

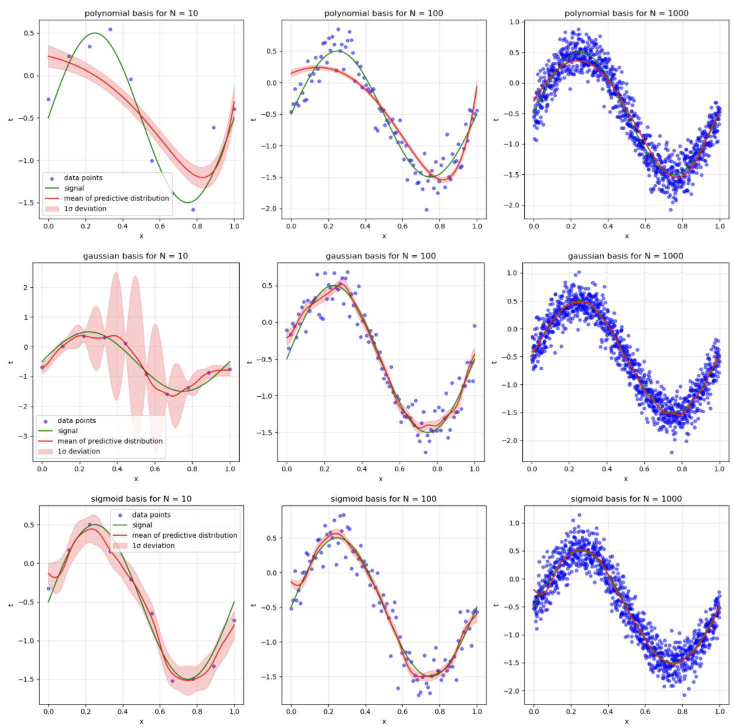

Figure 7:

- Higher risk of overfitting
- Models potentially capturing noise rather than underlying patterns

At moderate samples ($N = 100$), we achieve:

- Better balance between bias and variance
- More reasonable uncertainty estimates

Large samples ($N = 1000$) yield:

- Tight confidence intervals
- Consistent predictions across all basis functions

The χ^2/n_{dof} values approaching 1.0 for all three models suggest comparable performance, while maintaining distinct strengths in different scenarios. This highlights a fundamental machine learning principle: model selection should be guided by both data characteristics and prediction task requirements.

Strategies for Reliable Predictive Models

For obtaining reliable predictive models, several strategies prove effective:

1. Data quality and quantity must align with model complexity. More sophisticated basis functions require larger datasets to avoid overfitting.
2. Cross-validation facilitates appropriate basis function selection and parameter tuning, ensuring models generalize effectively to unseen data.
3. The Bayesian approach provides crucial uncertainty quantification, identifying regions of potentially less reliable predictions.
4. Monitoring both in-sample and out-of-sample performance enables early detection of overfitting.

Machine Learning Context

In the broader context of machine learning, this analysis demonstrates key concepts:

- The bias-variance tradeoff
- Importance of feature engineering through basis function selection
- Role of model complexity in generalization

The Bayesian framework naturally incorporates regularization, mitigating overfitting while providing uncertainty estimates. This comprehensive approach ensures robust and reliable predictive modeling across various scenarios.